



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/072,683	02/08/2002	Nir Zuk	0023-0209	2532

44987 7590 11/23/2007
HARRITY SNYDER, LLP
11350 Random Hills Road
SUITE 600
FAIRFAX, VA 22030

EXAMINER

NGUYEN, MINH DIEU T

ART UNIT	PAPER NUMBER
----------	--------------

2137

MAIL DATE	DELIVERY MODE
-----------	---------------

11/23/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/072,683
Filing Date: February 08, 2002
Appellant(s): ZUK ET AL.

MAILED

NOV 23 2007

Technology Center 2100

Glenn Snyder
Registration Number: 41,428
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 8/28/2007 appealing from the Office
action mailed 3/28/2007.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,499,107	Gleichauf et al.	12-2002
6,324,656	Gleichauf et al.	11-2001
6,253,321	Nikander et al.	6-2001
2003/0105976	Copeland, III	6-2003
2004/0258073	Alexander et al.	12-2004
6,453,345	Trcka et al.	9-2002

Navarro, "A Partial Deterministic Automaton for Approximate String Matching",
1997, Department of Computer Science, University of Chile.

Navarro et al., "Improving an Algorithm for Approximate Pattern Matching", 1998,
Department of Computer Science, University of Chile.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all
obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-7, 10, 12-18, 21, 23-25, 27, 31-33, 35, 37-38 and 40-41 are rejected
under 35 U.S.C. 103(a) as being unpatentable over Gleichauf et al. (6,499,107) and

Gleichauf et al. (6,324,656) in view of Nikander et al. (6,253,321) in view of Copeland, III (2003/0105976) and further in view of Alexander et al. (2004/0258073).

a) As to claims 1 and 24, Gleichauf discloses a method and system for adaptive network security using intelligent packet analysis comprising reassembling a plurality of TCP packets in the network traffic into a TCP stream, Gleichauf implicitly discloses this limitation (i.e. TCP stream reassembly) (col. 6, lines 39-40), to make it even clearer, the examiner takes official notice that use of reassembling TCP packets into a TCP stream is quite well known in data communications network. Data traveling over an IP network is always broken up into packets, the IP protocol adds information to each packet so that the routers along the network know where the data came and where it is going, the packets may be received out of order, or not, and are reassembled in the proper order at the destination computer; inspecting the TCP stream to detect information indicative of a security breach (col. 3, lines 1-4), wherein inspecting the TCP stream to detect information indicative of a security breach (col. 2, lines 50-55) comprises storing a plurality of protocol specifications supported by the network in a protocol database; and querying the protocol database to determine whether the plurality of TCP packets are compliant with one or more of the plurality of protocol specifications in the protocol database (col. 6, lines 31-33; col. 8, lines 20-35).

Gleichauf (6,324,656) also discloses inspecting the TCP stream to detect information indicative of a security breach comprises storing a plurality of protocol specifications supported by the network in a protocol database; and querying the protocol database to determine whether the plurality of TCP packets are compliant with

one or more of the plurality of protocol specifications in the protocol database (Fig. 3B; col. 6, lines 32 – col. 7, line 5).

Gleichauf does not explicitly disclose dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches.

Nikander is relied on for the teaching of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches (col. 4, lines 41-45).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches in the system of Gleichauf, as Nikander teaches so as to effectively manage communications data.

Gleichauf (6,499,107 and 6,324,656) and Nikander do not specifically disclose grouping the plurality of TCP packets into packet flows and sessions; storing the packet flows in packet flow descriptors and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream. Copeland is relied on for the teaching of grouping the plurality of TCP packets into packet flows and sessions (paragraphs 0039; 0050); storing the packet flows in

packet flow descriptors (paragraph 0050) and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream (paragraphs 0051, 005, 0081-0083, 0172). It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of grouping the plurality of TCP packets into packet flows and sessions in the system of Gleichauf and Nikander, as Copeland teaches so as to effectively determine if the traffic data appears to be legitimate or possible suspicious activity.

The combination of Gleichauf (6,499,107 and 6,324,656), Nikander and Copeland is silent on the capability of having the packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type.

Alexander is relied on for the teaching of discloses packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (i.e. computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (page 3, paragraph [0027])).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type in the system of Gleichauf (6,499,107 and 6,324,656), Nikander and Copeland as Alexander teaches so as to effectively performing packet filtering.

b) As to claims 2, 12 and 15, Gleichauf discloses inspecting the TCP stream to detect information indicative of security breaches comprising inspecting the TCP stream for protocol irregularities (col. 6, lines 36-42).

c) As to claims 3, 13, and 16-17, Gleichauf discloses inspecting the TCP to detect information indicative of a security breach comprising searching the TCP stream for attack signatures (col. 1, lines 29-31).

d) As to claims 4, 31 and 35, Gleichauf discloses searching the TCP stream for attack signatures comprises using stateful signature detection (col. 6, lines 45-52).

e) As to claims 5, 14 and 33, Gleichauf discloses inspecting the TCP stream to detect information indicative of a security breach using a plurality of network intrusion detection methods (col. 6, lines 66-67).

f) As to claim 6, Gleichauf discloses the plurality of network intrusion detection method comprises at least protocol anomaly detection (col. 6, lines 36-42).

g) As to claim 7, Gleichauf discloses the plurality of network intrusion detection methods comprises at least signature detection (col. 6, lines 43-45).

h) As to claim 10, Copeland discloses searching the packet flow descriptors for traffic signatures and inspecting the TCP stream comprises searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream (page 6, paragraph [0070]).

i) As to claims 18 and 27, Gleichauf discloses a method and system for adaptive network security using intelligent packet analysis comprising reassembling a plurality of TCP packets in the network traffic into a TCP stream, Gleichauf implicitly

discloses this limitation (i.e. TCP stream reassembly) (col. 6, lines 39-40), to make it even clearer, the examiner takes official notice that use of reassembling TCP packets into a TCP stream is quite well known in data communications network. Data traveling over an IP network is always broken up into packets, the IP protocol adds information to each packet so that the routers along the network know where the data came and where it is going, the packets may be received out of order, or not, and are reassembled in the proper order at the destination computer; inspecting the TCP stream to detect information indicative of a security breach (col. 3, lines 1-4).

Gleichenhauf does not explicitly disclose dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches.

Nikander is relied on for the teaching of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches (col. 4, lines 41-45).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches in the system of Gleichenhauf, as Nikander teaches so as to effectively manage communications data.

Gleichauf and Nikander do not expressly disclose grouping the plurality of TCP packets into packet flows and sessions, wherein grouping the plurality of TCP packets into packet flows and sessions comprises storing the packet flows and sessions in a hash table.

Copeland discloses a flow-based intrusion detection system for detecting intrusions in computer communication networks comprising grouping the plurality of TCP packets into packet flows and sessions (Fig. 1, elements "FLOW F1-FLOW F4"; page 5, paragraph [0058]; Fig. 3), wherein grouping the plurality of TCP packets into packet flows and sessions comprises storing the packet flows and sessions in a hash table (page 9, paragraph [0107]), wherein inspecting the TCP stream to detect information indicative of a security breach comprises storing the packet flows in packet flow descriptors (paragraph 0050) and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream (paragraphs 0051, 005, 0081-0083, 0172).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of grouping the plurality of TCP packets into packet flows and sessions, wherein grouping the plurality of TCP packets into packet flows and sessions comprises storing the packet flows and sessions in a hash table, wherein inspecting the TCP stream to detect information indicative of a security breach comprises storing the packet flows in packet flow descriptors and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream in the system of Gleichauf and Nikander, as

Copeland teaches so as to effectively determine if the traffic data appears to be legitimate or possible suspicious activity.

The combination of Gleichauf, Nikander and Copeland is silent on the capability of having the packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type.

Alexander is relied on for the teaching of discloses packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (i.e. computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (page 3, paragraph [0027])).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type in the system of Gleichauf, Nikander and Copeland as Alexander teaches so as to effectively performing packet filtering.

j) As to claims 21 and 38, Gleichauf discloses searching the TCP stream for attack signatures comprises querying the signatures database to determine whether there are matching signatures in the TCP stream (col. 6, lines 45-52; col. 5, lines 36-42).

l) As to claims 23 and 25, Gleichauf discloses reconstructing the plurality of TCP packets from a plurality of packet fragments (col. 6, lines 39-40).

m) As to claim 32, Copeland discloses a traffic signature detection software module for searching the packet flow descriptors for traffic signatures (page 4, paragraphs [0047-0051]).

n) As to claim 37, Gleichauf (6,324,656) discloses the protocol specifications comprise specifications of one or more of TCP protocol, HTTP protocol, SMTP protocol, FTP protocol, NETBIOS protocol, IMAP protocol, POP3 protocol, TELNET protocol, IRC protocol, RSH protocol, REXEC protocol, and RCMD protocol (Fig. 3B).

o) As to claim 40, Gleichauf discloses a routine for collecting a plurality of security logs and alarms recording information about security breaches found in the TCP stream (col. 7, lines 1-5); a routine for storing a network security policy identifying the network traffic to inspect and a plurality of network attacks to be detected and prevented (col. 5, lines 33-42); a routine for distributing the network security policy to one or more gateway points in the network (Fig. 2, element 20) and a routine for updating the protocol database and the signatures database (col. 9, lines 7-13).

p) As to claim 41, Copeland discloses the system further comprising a graphical user interface comprising a routine for displaying network security information to network security administrators; and a routine for specifying a network security policy (page 11, paragraph [0182]).

Claims 22 and 39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gleichauf et al. (6,499,107) and Gleichauf et al. (6,324,656) in view of Nikander et al. (6,253,321).

Gleichauf discloses a method and system for adaptive network security using intelligent packet analysis comprising reassembling a plurality of TCP packets in the network traffic into a TCP stream, Gleichauf implicitly discloses this limitation (i.e. TCP stream reassembly) (col. 6, lines 39-40), to make it even clearer, the examiner takes official notice that use of reassembling TCP packets into a TCP stream is quite well known in data communications network. Data traveling over an IP network is always broken up into packets, the IP protocol adds information to each packet so that the routers along the network know where the data came and where it is going, the packets may be received out of order, or not, and are reassembled in the proper order at the destination computer; inspecting the TCP stream to detect information indicative of a security breach (col. 3, lines 1-4), querying a signatures database to determine whether there are matching signatures in the TCP stream (col. 6, lines 45-52; col. 5, lines 36-42).

Gleichauf does not explicitly disclose dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches.

Nikander is relied on for the teaching of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and

forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches (col. 4, lines 41-45).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches in the system of Gleichauf, as Nikander teaches so as to effectively manage communications data.

Gleichauf and Nikander do not expressly disclose using deterministic finite automata for pattern matching when querying a signatures database to determine whether there are matching signatures in the TCP stream.

The examiner takes official notice that use of deterministic finite automaton for providing a pattern matching is well known in the theory of computation.

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of deterministic finite automaton for providing a pattern matching is well known in the theory of computation in the system of Gleichauf and Nikander so as to effectively implementing pattern matching.

Claims 42-46, 49-50 and 52-69 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gleichauf et al. (6,499,107) in view of Nikander et al. (6,253,321) in

view of Trcka et al. (6,453,345) in view of Copeland, III (2003/0105976) and further in view of Alexander et al. (2004/0258073).

a) As to claims 42 and 57, Gleichauf discloses a method and system for adaptive network security using intelligent packet analysis comprising reassembling a plurality of TCP packets in the network traffic into a TCP stream, Gleichauf implicitly discloses this limitation (i.e. TCP stream reassembly) on col. 6, lines 39-40, to make it even clearer, the examiner takes official notice that use of reassembling TCP packets into a TCP stream is quite well known in data communications network. Data traveling over an IP network is always broken up into packets, the IP protocol adds information to each packet so that the routers along the network know where the data came and where it is going, the packets may be received out of order, or not, and are reassembled in the proper order at the destination computer; inspecting the TCP stream to detect information indicative of security breaches (col. 3, lines 1-4), wherein inspecting the TCP stream to detect information indicative of a security breach (col. 2, lines 50-55) comprises storing a plurality of protocol specifications supported by the network in a protocol database; and querying the protocol database to determine whether the plurality of TCP packets are compliant with one or more of the plurality of protocol specifications in the protocol database (col. 6, lines 31-33; col. 8, lines 20-35).

Gleichauf (6,324,656) also discloses inspecting the TCP stream to detect information indicative of a security breach comprises storing a plurality of protocol specifications supported by the network in a protocol database; and querying the protocol database to determine whether the plurality of TCP packets are compliant with

one or more of the plurality of protocol specifications in the protocol database (Fig. 3B; col. 6, lines 32 – col. 7, line 5).

Gleichen does not disclose dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of a security breach and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of a security breach.

Nikander discloses dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of a security breach and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of a security breach (col. 4, lines 41-45).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of a security breach and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of a security breach in the system of Gleichen, as Nikander teaches so as to effectively manage communications data.

Gleichen and Nikander do not disclose a central management server and a graphical user interface.

Trcka discloses a network security and surveillance system comprising a central management center (col. 15, lines 13-21; Fig. 8, element 64) to control the network intrusion detection and prevention sensor and a graphical user interface for configuring the network intrusion detection and prevention sensor (col. 13, lines 50-65).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ to use of having a central management server to control the network intrusion detection and prevention sensor and a graphical user interface for configuring the network intrusion detection and prevention sensor (col. 13, lines 50-65) in the system of Gleichauf and Nikander as Trcka teaches so as to detect and protect against security breaches, network failures and other types of data compromising events (col. 1, lines 10-15).

Gleichauf, Nikander and Trcka do not specifically disclose grouping the plurality of TCP packets into packet flows and sessions; storing the packet flows in packet flow descriptors and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream. Copeland is relied on for the teaching of grouping the plurality of TCP packets into packet flows and sessions (paragraphs 0039; 0050); storing the packet flows in packet flow descriptors (paragraph 0050) and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream (paragraphs 0051, 005, 0081-0083, 0172). It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of grouping the plurality of TCP packets into packet flows and sessions, wherein grouping the plurality of TCP packets into packet flows and sessions comprises storing the packet flows and sessions in a hash table in the system of Gleichauf, Nikander and Trcka, as Copeland teaches so as to effectively determine if the traffic data appears to be legitimate or possible suspicious activity.

The combination of Gleichauf, Nikander, Trcka and Copeland is silent on the capability of having the packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type.

Alexander is relied on for the teaching of discloses packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (i.e. computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type (page 3, paragraph [0027])).

It would have been obvious to one of ordinary skill in the art at the time of the invention to employ the use of computing a hash value from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type in the system of Gleichauf, Nikander, Trcka and Copeland as Alexander teaches so as to effectively performing packet filtering.

b) As to claims 46 and 59, Nikander discloses dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of a security breach (col. 4, lines 41-45).

c) As to claims 50, 54, 66 and 69, Gleichauf discloses searching the TCP stream for attack signatures comprises using stateful signature detection (col. 6, lines 45-52).

d) As to claims 52 and 67, Gleichauf discloses inspecting the TCP stream to detect information indicative of a security breach using a plurality of network intrusion detection methods (col. 6, lines 66-67).

e) As to claim 49, 53, 65 and 68, Gleichauf discloses the plurality of network intrusion detection method comprises at least protocol anomaly detection (col. 6, lines 36-42).

f) As to claims 45 and 58, Gleichauf discloses reconstructing the plurality of TCP packets from a plurality of packet fragments (col. 6, lines 39-40).

g) As to claims 55, 60 and 63-64, Gleichauf discloses a routine for collecting a plurality of security logs and alarms recording information about security breaches found in the TCP stream (col. 7, lines 1-5); a routine for storing a network security policy identifying the network traffic to inspect and a plurality of network attacks to be detected and prevented (col. 5, lines 33-42); a routine for distributing the network security policy to one or more gateway points in the network (Fig. 2, element 20) and a routine for updating the protocol database and the signatures database (col. 9, lines 7-13).

h) As to claims 56, and 61-62, Copeland discloses the system further comprising a graphical user interface comprising a routine for displaying network security information to network security administrators; and a routine for specifying a network security policy (page 11, paragraph [0182]).

i) As to claim 43, Gleichauf discloses the network intrusion detection and prevention sensor is placed inside a firewall (col. 4, lines 47-49).

j) As to claim 44, Gleichauf discloses the network intrusion detection and prevention sensor is placed outside a firewall (col. 5, lines 24-27).

(10) Response to Argument

a) Appellant, on pages 11-16 of the brief, argues that the Final Office Action has not provided any statement regarding why it would have been obvious to combine Gleichauf '107 and Gleichauf '656; that it would not have been obvious to combine the unrelated references (i.e. Gleichauf '107, Gleichauf '656, Nikander et al., Copeland, III and Alexander et al.).

Gleichauf '656 is incorporated by reference in Gleichauf '107 (Gleichauf '107: col. 6, lines 15-20) and Gleichauf '656 is cited to further clearly address the limitations (i.e. storing a plurality of protocol specifications supported by the network in a protocol database and querying the protocol database to determine whether the plurality of TCP packets are compliant with one or more of the plurality of protocol specifications in the protocol database) that has been disclosed by Gleichauf '107.

Alexander is relied on for the teaching of having the packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type. Alexander is cited for providing specific detail on the 5 tuples (i.e. a source IP address, a destination IP address, a source port, a destination port and a protocol type) that Gleichauf '107 briefly discloses as checksum verification (IP, TCP, UDP, ICMP, etc.) (Gleichauf '107: col. 6, lines 38-39). TCP and UDP header contain source port and

destination port and like TCP, UDP uses IP for addressing and routing purposes, wherein IP header contains source IP address, destination IP address and protocol type. As such, it is proper to combine Alexander with Gleichauf '107 and Gleichauf '656 to implement a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type so as to effectively monitor network traffic (Gleichauf '107: col. 2, lines 44-55).

Nikander is relied on for the teaching of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches.

Copeland is relied on for the teaching of grouping the plurality of TCP packets into packet flows and sessions; storing the packet flows in packet flow descriptors and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream.

Both Nikander (i.e. a data processing system implements a security protocol based on processing data in packets, Nikander: Abstract) and Copeland (i.e. data packets are processed and assigned to various flows, then the flow statistics are analyzed to determine if the flow appears to be legitimate traffic or possible suspicious activity, Copeland: 0015) are related to packet security. Furthermore, Nikander discloses security association for securely processing packets between entities and secure policy rules might specify a set of selectors (source IP address, destination IP address, subnet, protocol, source port and destination port), Nikander: col. 1, lines 31-

47) and Copeland discloses structures of TCP/IP and UDP packet headers (Copeland: Fig. 2) and the use of hashing (Copeland: 0107, 0193). As such, it is proper to combine Nikander, Copeland with Gleichauf '107, Gleichauf '656 and Alexander et al. so as to effectively and securely process packet data over the network.

b) Appellant, on pages 11-16 of the brief, argues that the cited documents, Navarro 1997 and Navarro 1998 do not disclose the use of deterministic finite automata in the manner recited in claim 22.

Gleichauf '107 discloses querying a signatures database to determine whether there are matching signatures in the TCP stream (Gleichauf '107: col. 6, lines 45-52; col. 5, lines 36-42). However, Gleichauf is silent on the capability of using deterministic finite automata (DFA) for pattern matching. Navarro 1997 and 1998 disclose implementing pattern matching using DFA (Navarro 1997: Abstract, pages 2-3; Navarro 1998: page 2, 3rd paragraph; pages 6-7 and 10-15). As such, the combination of cited documents addresses limitations in claim 22.

c) Appellant, on pages 19-22 of the brief, argues that it would not have been obvious to combine the unrelated references (i.e. Gleichauf '107, Gleichauf '656, Nikander et al., Trcka et al., Copeland, III and Alexander et al.).

Gleichauf '656 is incorporated by reference in Gleichauf '107 (Gleichauf '107: col. 6, lines 15-20) and Gleichauf '656 is cited to further clearly address the limitations (i.e. storing a plurality of protocol specifications supported by the network in a protocol

database and querying the protocol database to determine whether the plurality of TCP packets are compliant with one or more of the plurality of protocol specifications in the protocol database) that has been disclosed by Gleichauf '107.

Alexander is relied on for the teaching of having the packet flow descriptors being addressed by a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type. Alexander is cited for providing specific detail on the 5 tuples (i.e. a source IP address, a destination IP address, a source port, a destination port and a protocol type) that Gleichauf '107 briefly discloses as checksum verification (IP, TCP, UDP, ICMP, etc.) (Gleichauf '107: col. 6, lines 38-39). TCP and UDP header contain source port and destination port and like TCP, UDP uses IP for addressing and routing purposes, wherein IP header contains source IP address, destination IP address and protocol type. As such, it is proper to combine Alexander with Gleichauf '107 and Gleichauf '656 to implement a hash value computed from a 5-tuple comprising a source IP address, a destination IP address, a source port, a destination port and a protocol type so as to effectively monitor network traffic (Gleichauf '107: col. 2, lines 44-55).

Nikander is relied on for the teaching of dropping a TCP packet from the TCP stream if the TCP stream contains information indicative of security breaches and forwarding a TCP packet to a network destination if the TCP stream does not contain information indicative of security breaches.

Trcka is relied on for the teaching of a central management server to control the network intrusion detection and prevention sensor and a graphical user interface for configuring the network intrusion detection and prevention sensor.

Copeland is relied on for the teaching of grouping the plurality of TCP packets into packet flows and sessions; storing the packet flows in packet flow descriptors and searching for a network attack identifier in the TCP stream based on the packet flow descriptors and sessions associated with the TCP stream.

Nikander (i.e. a data processing system implements a security protocol based on processing data in packets, Nikander: Abstract), Trcka (i.e. a network security and surveillance system monitors and records the traffic present on the network, Trcka: Abstract) and Copeland (i.e. data packets are processed and assigned to various flows, then the flow statistics are analyzed to determine if the flow appears to be legitimate traffic or possible suspicious activity, Copeland: 0015) are all related to packet security. Furthermore, Nikander discloses security association for securely processing packets between entities and secure policy rules might specify a set of selectors (source IP address, destination IP address, subnet, protocol, source port and destination port), Nikander: col. 1, lines 31-47), Trcka discloses filtering out the bad packets which has checksum error (Trcka: col. 6, lines 40-56) and Copeland discloses structures of TCP/IP and UDP packet headers (Copeland: Fig. 2) and the use of hashing (Copeland: 0107, 0193). As such, it is proper to combine Nikander, Trcka, Copeland, with Gleichauf '107, Gleichauf '656 and Alexander et al. so as to effectively and securely process packet data over the network.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,




Minh Dieu Nguyen

Patent Examiner - *Granted Temporary Full Signatory Authority*

Art Unit 2137

Conferees:


Gilberto Barron, Jr.
Supervisory Patent Examiner
Art Unit 2132

/Benjamin Lanier/
Benjamin E. Lanier
Primary Examiner
Art Unit 2132